

[illegible]

[illegible]

```
0001 0 MODULE RPG$SQRT ( XTITLE 'Get square root'  
0002 0 IDENT = '1-002' ! file: RPG$SQRT.B32 EDIT:DG1002  
0003 0 ) =  
0004 1 BEGIN  
0005 1  
0006 1 *****  
0007 1 *  
0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *  
0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *  
0010 1 * ALL RIGHTS RESERVED. *  
0011 1 *  
0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *  
0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *  
0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *  
0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *  
0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *  
0017 1 * TRANSFERRED. *  
0018 1 *  
0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *  
0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *  
0021 1 * CORPORATION. *  
0022 1 *  
0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *  
0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *  
0025 1 *  
0026 1 *****  
0027 1  
0028 1  
0029 1  
0030 1  
0031 1 ++  
0032 1 FACILITY: RPGII SUPPORT  
0033 1  
0034 1 ABSTRACT  
0035 1  
0036 1 This routine supports the RPG SQRT opcode.  
0037 1  
0038 1  
0039 1 ENVIRONMENT: Vax-11 User Mode  
0040 1  
0041 1 AUTHOR: Debess Grabazs, CREATION DATE: 8-Feb-1983  
0042 1  
0043 1 MODIFIED BY:  
0044 1  
0045 1 1-001 - Original. DG 8-Feb-1983  
0046 1 1-002 - Error RPG$_INVDATTYP changed to RPG$_INVARG. DG 11-Jul-1983  
0047 1 --  
0048 1  
0049 1 !<BLF/PAGE>
```


| | | | | |
|-----|------|---|----------------------------|--|
| 51 | 0050 | 1 | %SBTTL 'Declarations' | |
| 52 | 0051 | 1 | + PROLOGUE FILE: | |
| 53 | 0052 | 1 | - | |
| 54 | 0053 | 1 | | |
| 55 | 0054 | 1 | | |
| 56 | 0055 | 1 | REQUIRE 'RTLIN:RPGPROLOG'; | ! Switches, PSECTs, macros, |
| 57 | 0120 | 1 | | ! linkages and LIBRARYs |
| 58 | 0121 | 1 | | |
| 59 | 0122 | 1 | + LINKAGES | |
| 60 | 0123 | 1 | NONE | |
| 61 | 0124 | 1 | - | |
| 62 | 0125 | 1 | | |
| 63 | 0126 | 1 | | |
| 64 | 0127 | 1 | + TABLE OF CONTENTS: | |
| 65 | 0128 | 1 | - | |
| 66 | 0129 | 1 | | |
| 67 | 0130 | 1 | FORWARD ROUTINE | |
| 68 | 0131 | 1 | RPG\$SORT : NOVALUE ; | |
| 69 | 0132 | 1 | | |
| 70 | 0133 | 1 | | |
| 71 | 0134 | 1 | + INCLUDE FILES | |
| 72 | 0135 | 1 | NONE | |
| 73 | 0136 | 1 | - | |
| 74 | 0137 | 1 | | |
| 75 | 0138 | 1 | | |
| 76 | 0139 | 1 | + MACROS | |
| 77 | 0140 | 1 | NONE | |
| 78 | 0141 | 1 | - | |
| 79 | 0142 | 1 | | |
| 80 | 0143 | 1 | | |
| 81 | 0144 | 1 | + EQUATED SYMBOLS | |
| 82 | 0145 | 1 | NONE | |
| 83 | 0146 | 1 | - | |
| 84 | 0147 | 1 | | |
| 85 | 0148 | 1 | | |
| 86 | 0149 | 1 | + EXTERNAL REFERENCES | |
| 87 | 0150 | 1 | - | |
| 88 | 0151 | 1 | | |
| 89 | 0152 | 1 | | |
| 90 | 0153 | 1 | EXTERNAL ROUTINE | |
| 91 | 0154 | 1 | COB\$CVTID_R7: JSB_67 | ! Convert CIT to D floating |
| 92 | 0155 | 1 | COB\$CVTLI_R8: JSB_678 | ! Convert long to CIT (with scaling) |
| 93 | 0156 | 1 | COB\$CVTPD_R9: JSB_6789 | ! Convert packed to D floating |
| 94 | 0157 | 1 | COB\$CVTRDP_R9: JSB_6789 | ! Convert D floating to packed |
| 95 | 0158 | 1 | COB\$CVTWI_R8: JSB_678 | ! Convert word to CIT (with scaling) |
| 96 | 0159 | 1 | LIB\$STOP | ! Stop execution via signalling |
| 97 | 0160 | 1 | MTH\$DSQRT_R5: JSB_D; | ! Square root of D floating |
| 98 | 0161 | 1 | | |
| 99 | 0162 | 1 | EXTERNAL LITERAL | |
| 100 | 0163 | 1 | MTH\$_SQUROONEG, | ! Square root of negative number |
| 101 | 0164 | 1 | RPG\$_INVARG; | ! Invalid data type |
| 102 | 0165 | 1 | | |
| 103 | 0166 | 1 | EXTERNAL | |
| 104 | 0167 | 1 | LIB\$AB_CVTTP_O, | ! Table for convert trailing to packed |
| 105 | 0168 | 1 | RPG\$BTZ; | ! Table for translate blank to zero |

```
107 0169 1 %SBTTL 'RPG$SQRT - Get square root'
108 0170 1 GLOBAL ROUTINE RPG$SQRT(
109 0171 1     FLAGS,                                ! Translation flag
110 0172 1     NUMBER: REF BLOCK[,BYTE],           ! Argument for square root operation
111 0173 1     RESULT: REF BLOCK[,BYTE]           ! Result of square root operation
112 0174 1     ): NOVALUE=
113 0175 1
114 0176 1 ++      FUNCTIONAL DESCRIPTION
115 0177 1
116 0178 1      This routine supports the RPG SQRT opcode. It is
117 0179 1      called once by the compiled code for each occurrence
118 0180 1      of the SQRT opcode for scalars, or once for each
119 0181 1      element of an array.
120 0182 1      It accepts an input number of word, long, packed, or
121 0183 1      right overpunched numeric data type; and outputs a
122 0184 1      packed result.
123 0185 1
124 0186 1      CALLING SEQUENCE:
125 0187 1
126 0188 1          CALL RPG$SQRT (flags.rl.v, number.rx.ds, result.wp.ds)
127 0189 1
128 0190 1      FORMAL PARAMETERS:
129 0191 1
130 0192 1          flags      longword integer - bit 1 set if blanks in
131 0193 1                      overpunched numeric field should be treated
132 0194 1                      as equivalent to zeroes; otherwise the
133 0195 1                      translation is not done.
134 0196 1
135 0197 1          number      address of descriptor of argument for square
136 0198 1                      root operation.
137 0199 1                      The allowable data types are word, long,
138 0200 1                      packed, and right overpunched numeric.
139 0201 1
140 0202 1          result      address of descriptor of result of the square
141 0203 1                      root operation.
142 0204 1                      The allowable data type is packed.
143 0205 1
144 0206 1      IMPLICIT INPUTS:
145 0207 1
146 0208 1          NONE
147 0209 1
148 0210 1      IMPLICIT OUTPUTS:
149 0211 1
150 0212 1          NONE
151 0213 1
152 0214 1      COMPLETION CODES:
153 0215 1
154 0216 1          SSS_NORMAL
155 0217 1
156 0218 1      SIDE EFFECTS:
157 0219 1
158 0220 1          If NUMBER is negative, the result field is set to zero and the
159 0221 1          error MTH$_SQURONEG is signalled.
160 0222 1
161 0223 1      --
162 0224 1
163 0225 1
```

RPG\$SQRT
1-002

Get square root
RPG\$SQRT - Get square root

: 164

0226 1 !<BLF/PAGE>

I 4
16-Sep-1984 02:19:11
14-Sep-1984 13:04:26

VAX-11 Bliss-32 V4.0-742
[RPGRTL.SRC]RPG\$SQRT.B32;1

Page 4
(3)

RPG
1-0

59
54
30


```
166 0227 1  
167 0228 2  
168 0229 2  
169 0230 2  
170 0231 2  
171 0232 2  
172 0233 2  
173 0234 2  
174 0235 2  
175 0236 2  
176 0237 2  
177 0238 2  
178 0239 2  
179 0240 2  
180 0241 2  
181 0242 2  
182 0243 2  
183 0244 2  
184 0245 2  
185 0246 2  
186 0247 2  
187 0248 2  
188 0249 2  
189 0250 2  
190 0251 2  
191 0252 2  
192 0253 2  
193 0254 2  
194 0255 2  
195 0256 2  
196 0257 2  
197 0258 2  
198 0259 2  
199 0260 2  
200 0261 2  
201 0262 2  
202 0263 2  
203 0264 2  
204 0265 2  
205 0266 2  
206 0267 2  
207 0268 2  
208 0269 2  
209 0270 2  
210 0271 2  
211 0272 2  
212 0273 2  
213 0274 2  
214 0275 2  
215 0276 2  
216 0277 2  
217 0278 2  
218 0279 2  
219 0280 2  
220 0281 2  
221 0282 2  
222 0283 2  
  
BEGIN  
  
LITERAL  
    BTZ_BIT = 2, ! Convert blanks to zeroes  
    MAX_PACKED_LEN = 15; ! Maximum allowed packed decimal number length  
  
LOCAL  
    D_VALUE: VECTOR[2], ! Input number converted to D floating  
    D_SQRT: VECTOR[2], ! D floating square root result  
    I_VALUE: VECTOR[12, BYTE], ! COBOL intermediate temporary  
    PACKED_LENGTH,  
    PACKED_NUMBER: VECTOR [MAX_PACKED_LEN/2 + 1, BYTE],  
    SCALE: ! Packed decimal number  
    ! Scale factor  
  
BUILTIN  
    CVTTP; ! Convert trailing to packed  
  
!+  
! Get the scale factor.  
!-  
SCALE = (IF .NUMBER[DSC$B_CLASS] EQL DSC$K_CLASS_SD  
    THEN .NUMBER[DSC$B_SCALE]  
    ELSE 0);  
  
!+  
! Convert the input number to D floating  
!-  
SELECTONE .NUMBER[DSC$B_DTYPE] OF  
    SET  
    [DSC$K_DTYPE_W]: ! Word  
    BEGIN  
        !+  
        ! Convert word to CIT to d floating  
        ! (so scale doesn't get lost).  
        !-  
        COB$CVTWI_R8 (.SCALE, .NUMBER[DSC$A_POINTER], I_VALUE);  
        COB$CVTID_R7 (I_VALUE, D_VALUE);  
    END;  
    [DSC$K_DTYPE_L]: ! Long  
    BEGIN  
        !+  
        ! Convert long to CIT to d floating  
        ! (so scale doesn't get lost).  
        !-  
        COB$CVTLI_R8 (.SCALE, .NUMBER[DSC$A_POINTER], I_VALUE);  
        COB$CVTID_R7 (I_VALUE, D_VALUE);  
    END;  
END;
```

RPG\$SQRT
1-002

Get square root
RPG\$SQRT - Get square root

K 4
16-Sep-1984 02:19:11
14-Sep-1984 13:04:26

VAX-11 Bliss-32 V4.0-742
[RPGRTL.SRC]RPG\$SQRT.B32;1

Page 6
(4)

```
223 0284 2 [DSC$K_DTYPE_P]: ! Packed
224 0285
225 0286 COB$CVTPD_R9 (.SCALE, .NUMBER[DSC$W_LENGTH], .NUMBER[DSC$A_POINTER], D_VALUE);
226 0287
227 0288 [DSC$K_DTYPE_NRO]: ! Right overpunched numeric
228 0289 BEGIN
229 0290
230 0291 IF (.FLAGS AND BTZ_BIT) NEQ 0
231 0292 THEN
232 0293 !+
233 0294 ! Translate blanks to zeroes if flag set.
234 0295 !-
235 0296 CH$TRANSLATE (RPG$BTZ, .NUMBER[DSC$W_LENGTH], .NUMBER[DSC$A_POINTER],
236 0297 0, .NUMBER[DSC$W_LENGTH], .NUMBER[DSC$A_POINTER]);
237 0298 !+
238 0299 ! Convert trailing to packed to d-floating.
239 0300 !-
240 0301 PACKED_LENGTH = MAX_PACKED_LEN;
241 0302 CVTTP (.NUMBER[DSC$W_LENGTH], .NUMBER[DSC$A_POINTER], LIB$AB_CVTTP_0, PACKED_LENGTH, PACKED_NUMBE
242 0303 COB$CVTPD_R9 (.SCALE, MAX_PACKED_LEN, PACKED_NUMBER, D_VALUE);
243 0304
244 0305 END;
245 0306 [OTHERWISE]:
246 0307
247 0308 LIB$STOP (RPG$INVARG);
248 0309
249 0310 TES;
250 0311
251 0312 !+
252 0313 ! Take the square root of the D-floating value and
253 0314 ! convert the result to the output data type (packed)
254 0315 !-
255 0316
256 0317
257 0318 MTH$DSQRT R5 (.D_VALUE[0], .D_VALUE[1]; D_SQRT[0], D_SQRT[1]);
258 0319 SCALE = (IF .RESULT[DSC$B_CLASS] EQL DSC$R_CLASS_SD
259 0320 THEN .RESULT[DSC$B_SCALE]
260 0321 ELSE 0);
261 0322 COB$CVTRDP_R9 (-.SCALE, D_SQRT, .RESULT[DSC$W_LENGTH], .RESULT[DSC$A_POINTER]);
262 0323
263 0324 END;
```

.TITLE RPG\$SQRT Get square root
.IDENT \1-002\

.EXTRN COB\$CVTID_R7, COB\$CVTLI_R8
.EXTRN COB\$CVTPD_R9, COB\$CVTRDP_R9
.EXTRN COB\$CVTWI_R8, LIB\$STOP
.EXTRN MTH\$DSQRT_R5, MTH\$SQUROONEG
.EXTRN RPG\$INVARG, LIB\$AB_CVTTP_0
.EXTRN RPG\$BTZ

.PSECT _RPG\$CODE, NOWRT, SHR, PIC, 2

OFFC 00000

.ENTRY RPG\$SQRT, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,-; 0170
R11

RPG\$SQRT
1-002

Get square root
RPG\$SQRT - Get square root

L 4
16-Sep-1984 02:19:11
14-Sep-1984 13:04:26

VAX-11 Bliss-32 V4.0-742
[RPGRTL.SRC]RPG\$SQRT.B32;1

Page 7
(4)

| | | | | | | | | | | | |
|-----------|----|----|----|----|----|----|-------|--------|--|--|------|
| | | | 5E | | 24 | C2 | 00002 | SUBL2 | #36, SP | | |
| | | | 5A | 08 | AC | D0 | 00005 | MOVL | NUMBER, R10 | | 0251 |
| | | | 09 | 03 | AA | 91 | 00009 | CMPB | 3(R10), #9 | | |
| | | | | | 06 | 12 | 0000D | BNEQ | 1\$ | | |
| | | | 5B | 08 | AA | 98 | 0000F | CVTBL | 8(R10), SCALE | | 0252 |
| | | | | | 02 | 11 | 00013 | BRB | 2\$ | | |
| | | | | | 5B | D4 | 00015 | CLRL | SCALE | | 0251 |
| | | | 50 | 02 | AA | 9A | 00017 | MOVZBL | 2(R10), R0 | | 0260 |
| | | | 07 | | 50 | 91 | 0001B | CMPB | R0, #7 | | 0262 |
| | | | | | 13 | 12 | 0001E | BNEQ | 3\$ | | |
| | | | 58 | 08 | AE | 9E | 00020 | MOVAB | I-VALUE, R8 | | 0269 |
| | | | 57 | 04 | AA | D0 | 00024 | MOVL | 4(R10), R7 | | |
| | | | 56 | | 5B | D0 | 00028 | MOVL | SCALE, R6 | | |
| | | | | | 00 | 16 | 0002B | JSB | COB\$CVTWI_R8 | | |
| | | | | | 16 | 11 | 00031 | BRB | 4\$ | | 0270 |
| | | | 08 | | 50 | 91 | 00033 | CMPB | R0, #8 | | 0273 |
| | | | | | 21 | 12 | 00036 | BNEQ | 5\$ | | |
| | | | 58 | 08 | AE | 9E | 00038 | MOVAB | I-VALUE, R8 | | 0280 |
| | | | 57 | 04 | AA | D0 | 0003C | MOVL | 4(R10), R7 | | |
| | | | 56 | | 5B | D0 | 00040 | MOVL | SCALE, R6 | | |
| | | | | | 00 | 16 | 00043 | JSB | COB\$CVTLI_R8 | | |
| | | | 57 | 1C | AE | 9E | 00049 | MOVAB | D-VALUE, R7 | | 0281 |
| | | | 56 | 08 | AE | 9E | 0004D | MOVAB | I-VALUE, R6 | | |
| | | | | | 00 | 16 | 00051 | JSB | COB\$CVTID_R7 | | |
| | | | | | 58 | 11 | 00057 | BRB | 10\$ | | 0260 |
| | | | 15 | | 50 | 91 | 00059 | CMPB | R0, #21 | | 0284 |
| | | | | | 0D | 12 | 0005C | BNEQ | 6\$ | | |
| | | | 59 | 1C | AE | 9E | 0005E | MOVAB | D-VALUE, R9 | | 0286 |
| | | | 58 | 04 | AA | D0 | 00062 | MOVL | 4(R10), R8 | | |
| | | | 57 | | 6A | 3C | 00066 | MOVZWL | (R10), R7 | | |
| | | | | | 2E | 11 | 00069 | BRB | 8\$ | | |
| | | | 13 | | 50 | 91 | 0006B | CMPB | R0, #19 | | 0288 |
| | | | | | 34 | 12 | 0006E | BNEQ | 9\$ | | |
| | | | | | 21 | E1 | 00070 | BBC | #33, FLAGS, 7\$ | | 0291 |
| 00000000G | 00 | 0D | 6C | | 6A | 2E | 00074 | MOVTC | (R10), @4(R10), #0, RPG\$BTZ, (R10), @4(R10) | | 0297 |
| | | 00 | 04 | | 6A | | 0007E | | | | |
| | | | 50 | | 0F | D0 | 00081 | MOVL | #15, PACKED LENGTH | | 0301 |
| | | | | | 6A | 26 | 00084 | CVTTP | (R10), @4(R10), LIB\$AB CVTTP_0, - | | 0302 |
| | | | | | 6E | | 0008E | | PACKED LENGTH, PACKED_NUMBER | | |
| | | | 59 | 1C | AE | 9E | 0008F | MOVAB | D-VALUE, R9 | | 0303 |
| | | | 58 | | 6E | 9E | 00093 | MOVAB | PACKED_NUMBER, R8 | | |
| | | | 57 | | 0F | D0 | 00096 | MOVL | #15, R7 | | |
| | | | 56 | | 5B | D0 | 00099 | MOVL | SCALE, R6 | | |
| | | | | | 00 | 16 | 0009C | JSB | COB\$CVTPD_R9 | | |
| | | | | | 0D | 11 | 000A2 | BRB | 10\$ | | 0260 |
| | | | | | 8F | DD | 000A4 | PUSHL | #RPG\$ INVARG | | 0308 |
| | | | | | 01 | FB | 000AA | CALLS | #1, LIB\$STOP | | |
| 00000000G | 00 | | 50 | 1C | AE | 7D | 000B1 | MOVQ | D-VALUE, R0 | | 0318 |
| | | | | | 00 | 16 | 000B5 | JSB | MTH\$DSQRT_R5 | | |
| | | | 14 | | 50 | 7D | 000BB | MOVQ | R0, D_SQRT | | |
| | | | | | 0C | AC | 000BF | MOVL | RESULT, R0 | | 0319 |
| | | | 50 | 0C | AC | D0 | 000BF | | | | |
| | | | 09 | 03 | A0 | 91 | 000C3 | CMPB | 3(R0), #9 | | |
| | | | | | 06 | 12 | 000C7 | BNEQ | 11\$ | | |
| | | | 5B | 08 | A0 | 98 | 000C9 | CVTBL | 8(R0), SCALE | | 0320 |
| | | | | | 02 | 11 | 000CD | BRB | 12\$ | | |
| | | | | | 5B | D4 | 000CF | CLRL | SCALE | | 0319 |
| | | | 57 | 14 | AE | 9E | 000D1 | MOVAB | D_SQRT, R7 | | 0322 |

| | | | | | |
|----|-----------|----|----------|--------|----------------|
| 56 | | 5B | CE 000D5 | MNEGL | SCALE, R6 |
| 59 | 04 | A0 | D0 000D8 | MOVL | 4(R0), R9 |
| 58 | | 60 | 3C 000DC | MOVZWL | (R0), R8 |
| | 00000000G | 00 | 16 000DF | JSB | COBS\$VTRDP_R9 |
| | | 04 | 000E5 | RET | |

0324

```
; Routine Size: 230 bytes,    Routine Base: _RPG$CODE + 0000
```

```

: 264      0325 1
: 265      0326 0 END ELUDOM

```

PSECT SUMMARY

| Name | Bytes | Attributes |
|------------|-------|---|
| _RPG\$CODE | 230 | NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2) |

Library Statistics

| File | Total | Symbols Loaded | Percent | Pages Mapped | Processing Time |
|---------------------------------------|-------|-------------------|---------|-----------------|--------------------|
| \$255\$DUA28:[SYSLIB]STARLET.L32:1 | 9776 | 10 | 0 | 581 | 00:00.9 |
| \$255\$DUA28:[RPGRTL.OBJ]RPGLIB.L32:1 | 54 | 4 | 7 | 9 | 00:00.1 |








































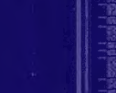




























































































































































COMMAND QUALIFIERS

```
; BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS$:RPGSQRT/OBJ=OBJ$:RPGSQRT MSRC$:RPGSQRT/UPDATE=(ENH$:RPGSQRT)
```

```
; Size:          230 code + 0 data bytes
; Run Time:      00:06.1
; Elapsed Time:  00:18.3
; Lines/CPU Min: 3217
; Lexemes/CPU-Min: 13430
; Memory Used:   91 pages
; Compilation Complete
```


0332 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|--|---|---|---|---|---|---|---|---|---|---|
|  | RPGMSGTX LIS |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  | DTE_DF03 MAP |  |  |  |  |  |  |  |  |  |  |
| RPGMOVE3 LIS |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | RPGSORT LIS |  |  |  |  |  |  |  |  |  |  |  |  |
|  | RPGOPEN LIS |  |  |  |  | RTPAD |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | CTDRIVER MAP |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | RTPAD MAP |  |  |  |  | RTPADMACS MAR |  |  |  |  |
| RPGMSGPTR LIS |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | RPGVECTOR LIS |  |  |  |  |  |  |  |  |  |
|  | RPGPRINT LIS |  |  |  |  |  |  |  |  |  |  |  |  |  |  |